

## WHAT IS CLAIMED IS:

- 1 1. A computer-implemented method for sharing one or more high-level language  
2 data structures between an assembly language program and a high-level language  
3 program, each data structure including one or more elements, comprising the steps of:  
4 determining storage requirements from high-level language definitions of the one  
5 or more data structures in an assembly language source program and removing the  
6 definitions from the assembly source program;  
7 determining memory addresses of the one or more data structures from memory  
8 allocation directives in the assembly language source program and removing the  
9 allocation directives from the assembly source program; and  
10 replacing references to elements of the data structures in the assembly source  
11 program with memory addresses.
- 1 2. The method of claim 1, further comprising:  
2 assembling the assembly source program into a first object code segment;  
3 compiling the high-level language program into a second object code segment;  
4 and  
5 linking the first and second object code segments.
- 1 3. The method of claim 1, wherein each allocation directive includes a reference to a  
2 data structure definition, a variable name, and an address.
- 1 4. The method of claim 3, wherein variable name includes an array size.
- 1 5. The method of claim 1, wherein a reference to an element of a data structure in  
2 the assembly source program includes a hierarchical specification of the element.

1 6. The method of claim 5, wherein a reference to an element of a data structure in  
2 the assembly source program includes one of a request for an address and a request for an  
3 offset address of the element.

1 7. The method of claim 5, wherein a reference to an element of a data structure in  
2 the assembly source program includes one of a request for an address of the element, a  
3 request for an offset address of the element, and a request for a size of the element.

1 8. The method of claim 7, further comprising replacing a reference to an element of  
2 a data structure that includes a request for a size of an element with the size of the  
3 element.

1 9. The method of claim 8, further comprising:  
2 assembling the assembly source program into a first object code segment;  
3 compiling the high-level language program into a second object code segment;  
4 and  
5 linking the first and second object code segments.

1 10. The method of claim 8, wherein each allocation directive includes a reference to a  
2 data structure definition, a variable name, and an address.

1 11. The method of claim 10, wherein variable name includes an array size.

1 12. The method of claim 5, further comprising:  
2 assembling the assembly source program into a first object code segment;  
3 compiling the high-level language program into a second object code segment;  
4 and  
5 linking the first and second object code segments.

1 13. The method of claim 5, wherein each allocation directive includes a reference to a  
2 data structure definition, a variable name, and an address.

- 1 15. An apparatus for sharing one or more high-level language data structures between  
2 an assembly language program and a high-level language program, each data structure  
3 including one or more elements, comprising:  
4 means for determining storage requirements from high-level language definitions  
5 of the one or more data structures in an assembly language source program and removing  
6 the definitions from the assembly source program;  
7 means for determining memory addresses of the one or more data structures from  
8 memory allocation directives in the assembly language source program and removing the  
9 allocation directives from the assembly source program; and  
10 means for replacing references to elements of the data structures in the assembly  
11 source program with memory addresses.